IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark Office

Help FAQ Terms IEEE Peer Review **Quick Links** » **Search Results**

Welcome to IEEE *Xplore*®

- ○- Home
- ○- What Can I Access?
- ○- Log-out

Tables of Contents

- ○- Journals & Magazines
- ○- Conference Proceedings
- ○- Standards

Search

- ○- By Author
- ○- Basic
- ○- Advanced

Member Services

- ○- Join IEEE
- ○- Establish IEEE Web Account
- ○- Access the IEEE Member Digital Library

Print Format

Your search matched **[0]** of **[966109]** documents.

You may refine your search by editing the current search expression or entering a new one the text box. Then click search Again.

(cost function) <and> (attribute of route) Search Again

OR

Use your browser's back button to return to your original search page.

Results:

No documents matched your query.

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark Office

Help FAQ Terms IEEE Peer Review Quick Links

» **Search Results**

Welcome to IEEE Xplore®

- ○ Home
- ○ What Can I Access?
- ○ Log-out

Tables of Contents

- ○ Journals & Magazines
- ○ Conference Proceedings
- ○ Standards

Search

- ○ By Author
- ○ Basic
- ○ Advanced

Member Services

- ○ Join IEEE
- ○ Establish IEEE Web Account
- ○ Access the IEEE Member Digital Library

Print Format

Your search matched **3** of **966109** documents.

A maximum of **3** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again**.

(cost factor) <and> (route) Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **A dynamic timeout scheme for wormhole routing networks**
Po-Chi Hu; Kleinrock, L.;
Communications, 1997. ICC 97 Montreal, 'Towards the Knowledge Millennium'. 1997 IEEE International Conference on , Volume: 3 , 8-12 June 1997
Page(s): 1406 -1410 vol.3

[Abstract] [PDF Full-Text (432 KB)] **IEEE CNF**

2 **Allocation of multiport memories for hierarchical data streams**
Lippens, P.E.R.; van Meerbergen, J.L.; Verhaegh, W.F.J.; van der Werf, A.;
Computer-Aided Design, 1993. ICCAD-93. Digest of Technical Papers., 1993 IEEE/ACM International Conference on , 7-11 Nov. 1993
Page(s): 728 -735

[Abstract] [PDF Full-Text (676 KB)] **IEEE CNF**

3 **Grooming of arbitrary traffic in SONET/WDM BLSRs**
Peng-Jun Wan; Calinescu, G.; Frieder, O.;
Selected Areas in Communications, IEEE Journal on , Volume: 18 Issue: 10 , Oct. 2000
Page(s): 1995 -2003

[Abstract] [PDF Full-Text (156 KB)] **IEEE JNL**

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark Office

Help FAQ Terms IEEE Peer Review

Quick Links

» **Search Results**

Welcome to IEEE *Xplore*®

- ○ Home
- ○ What Can I Access?
- ○ Log-out

Tables of Contents

- ○ Journals & Magazines
- ○ Conference Proceedings
- ○ Standards

Search

- ○ By Author
- ○ Basic
- ○ Advanced

Member Services

- ○ Join IEEE
- ○ Establish IEEE Web Account
- ○ Access the IEEE Member Digital Library

Print Format

Your search matched **158** of **966109** documents.

A maximum of **158** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again**.

(cost function) <and> (route) Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **Stability of a dynamic model for traffic networks**
Mounce, R.;
Intelligent Transportation Systems, 2002. Proceedings. The IEEE 5th International Conference on , 2002
Page(s): 795 -800

[Abstract] [PDF Full-Text (403 KB)] **IEEE CNF**

2 **On routing real-time multicast connections**
Matta, I.; Liang Guo;
Computers and Communications, 1999. Proceedings. IEEE International Symposium on , 6-8 July 1999
Page(s): 65 -71

[Abstract] [PDF Full-Text (568 KB)] **IEEE CNF**

3 **Hierarchical QoS routing in ATM networks based on MDP cost function**
Ben-Jye Chang; Ren-Hung Hwang;
Networks, 2000. (ICON 2000). Proceedings. IEEE International Conference on , 5-8 Sept. 2000
Page(s): 147 -151

[Abstract] [PDF Full-Text (456 KB)] **IEEE CNF**

4 **R uting for reliable manufacturing**
Huijbregts, E.P.; Hua Xue; Jess, J.A.G.;
Semiconductor Manufacturing, IEEE Transactions on , Volume: 8 Issue: 2 , May 1995
Page(s): 188 -194

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark Office

Help FAQ Terms IEEE Peer Review Quick Links » Abstract Plus

Welcome to IEEE Xplore®

- Home
- What Can I Access?
- Log-out

Tables of Contents

- Journals & Magazines
- Conference Proceedings
- Standards

Search

- By Author
- Basic
- Advanced

Member Services

- Join IEEE
- Establish IEEE Web Account
- Access the IEEE Member Digital Library

Print Format

SEARCH RESULTS [PDF Full-Text (628 KB)] PREVIOUS NEXT DOWNLOAD CITATION

Routing for reliable manufacturing

Huijbregts, E.P. Hua Xue Jess, J.A.G.
Philips Res. Lab., Eindhoven;

This paper appears in: **Semiconductor Manufacturing, IEEE Transactions on**

Publication Date: May 1995
On page(s): 188-194
Volume: 8, Issue: 2
ISSN: 0894-6507
References Cited: 14
CODEN: ITSMED
INSPEC Accession Number: 4965191

Abstract:

The impact of spot defects on the susceptibility for electrical failure of a net is analyzed. Based on this analysis, a general routing cost function is presented, in which the manufacturability of a net is taken into account in conjunction with traditional routing objectives. The new cost function, relating the process spot defects to the routing procedure has been implemented. Failure probabilities are analyzed for the benchmark layouts obtained by our routing tool using both the original cost function and the new cost function. The results show that the failure probability of a layout is significantly decreased if the spot defect mechanism is taken into account in the routing procedure, while the area of the layout is kept constant

Index Terms:

economics failure analysis integrated circuit layout integrated circuit manufacture integrated circuit reliability network routing benchmark layouts cost function electrical failure failure probabilities net process spot defects reliable manufacturing routing

Documents that cite this document

Select link to view other documents in the database that cite this one.

SEARCH RESULTS [PDF Full-Text (628 KB)] PREVIOUS NEXT DOWNLOAD CITATION

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Routing for Reliable Manufacturing

Ed P. Huijbregts, Hua Xue, and Jochen A. G. Jess

Abstract—**The impact of spot defects on the susceptibility for electrical failure of a net is analyzed. Based on this analysis, a general routing cost function is presented, in which the manufacturability of a net is taken into account in conjunction with traditional routing objectives. The new cost function, relating the process spot defects to the routing procedure has been implemented. Failure probabilities are analyzed for the benchmark layouts obtained by our routing tool using both the original cost function and the new cost function. The results show that the failure probability of a layout is significantly decreased if the spot defect mechanism is taken into account in the routing procedure, while the area of the layout is kept constant.**

I. Introduction

ROUTING a net is a "classical" topic in CAD for VLSI. The problem can be formalized as the Minimum Steiner Tree problem in an appropriate routing graph [6]:

Problem: minimum Steiner Tree,

Instance: a connected undirected graph $G = (V, E)$, also called routing graph, with edge cost function $\lambda : E \rightarrow \mathbf{R}_+$ and a net $N \subseteq V$, consisting of vertices to be connected,

Configurations: all edge-weighted trees,

Solutions: all Steiner trees for N in G, denoted as E_T, i.e., all trees of G connecting all vertices of N with all its leaves being vertices in N,

Minimize: $\lambda(T) = \Sigma_{e \in E_T} \lambda(e)$.

Many algorithms exist to solve the minimum Steiner tree problem, see [4] for an excellent overview. All of these algorithms will come up with significantly different routings if different cost functions are applied. Conventionally, the edge cost function $\lambda(e)$ is defined as the product of the distance d between two adjacent vertices and a control factor c, i.e., $\lambda(e) = cd$. Parameter c is used to adjust the edge weights in or between the different mask layers. For example, by setting a larger value of c for the poly layer and a smaller value of c for the metal layer, connections with high signal propagation speed can be obtained instead of a real shortest path in distance. Furthermore, by choosing different values for c for different routing directions, thus favoring certain directions, a routing style can be imposed. Summarizing, the traditional cost function can affect a routing in three aspects: the net length, the performance, and the routing style.

As process feature size keeps decreasing and IC chips are becoming more complex, chips are more sensitive to process disturbance. *Inductive Fault Analysis* (IFA) [9] reveals that close nets are likely to get shorted because of spot defects, the main local disturbance in fabrication processes. Therefore, from the point of view of defect analysis, the yield of a good routing depends not only on the net itself, but also on the environment of the net. In other words, the minimization of the cost of a net in terms of the net length is not the optimal solution if the failure possibility of the net is taken into account. Obviously, the proposed cost function $\lambda(e)$ does not adequately cover this issue.

The idea to relate the routing procedure to the process defects has been proposed in several papers [1], [2], [5], [8]. In [8], a channel router called Defect Tolerant Routing (DTR) was implemented to minimize the critical areas between the horizontal routing segments. Later on, the authors [1] tried to minimize the critical areas between both the horizontal segments and the vertical segments by searching for *valid gaps* in routing channels. It has also been proposed that layout be modified in order to minimize via count and critical areas on each layer for two layer routing channels [5]. In all papers, there are still quite a few drawbacks that make the routing results far from being effectively defect-tolerant. The main reasons are as follows.

1) Only spot defects causing extra material (bridges) are considered. Consequently, when the probability of bridge faults decreases by minimizing the critical area for bridges, the probability of open faults, caused by missing material, likely increases. This is a valid assumption since routers generally try to minimize the net length, and therefore the probability of open faults is minimized. Modifying such a 'minimum net length' layout in order to minimize the probability of bridge faults will usually result in longer net, and therefore the probability of opens increases.
2) Only the single layer defect model is used for modeling the spot defects. The fact that in addition missing material or extra material between mask layers will give rise to more bridges or opens is not taken into account.
3) The tradeoff between the increase of the number of vias (potential open sites) and the decrease of the critical area for bridges is not considered.
4) Only one defect size is considered. However, spot defects are distributed with random sizes in reality. To accurately model spot defects, it is important to take into account the defect size distribution.

In this paper, according to the defect size distribution and the process statistics, the failure probability of a net is analyzed.

Manuscript received May 1, 1994; revised October 12, 1994. This work was supported by Foundation F.O.M., The Netherlands, under Project EEL88.1428.

E. P. Huijbregts is with Philips Research Laboratories, Eindhoven, The Netherlands.

H. Xue is with Xilinx Inc., San Jose, CA 95124 USA.

J. A. G. Jess is with the Eindhoven University of Technology, Eindhoven, The Netherlands.

IEEE Log Number 9409816.

0894-6507/95$04.00 © 1995 IEEE

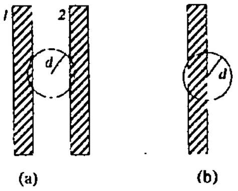


Fig. 1. (a) Fault type OE. (b) Fault type OM.

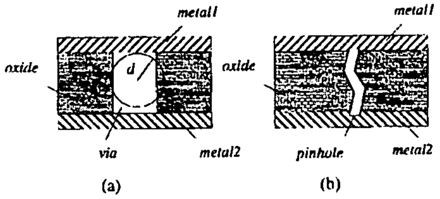


Fig. 2. (a) Fault type IE. (b) Fault type IM.

Based on the analysis, we propose a new edge cost function for the general routing problem. By applying the new cost function, a good tradeoff between the minimization of the net length and the minimization of the failure probability can be obtained for each net, which consequently leads to a better layout manufacturability. Part of this work has been published in [14].

II. Spot Defects

The functional failure of a chip is likely caused by spot defects [12]. The result of a spot contamination in a process step is either extra material or missing material at the place where the spot occurs [7]. A spot defect may either occur in one layer of the silicon structure, such as the metal layer or the poly layer, or somewhere between two layers, where it causes extra or missing oxide. We classify spot defects as follows.

1) *One layer extra material defects (OE)*: The defects may cause bridges between connection patterns in the same layer. For example, the spot defect with size d in the metal layer will result in a bridge between nets 1 and 2 as shown in Fig. 1(a).
2) *One layer missing material defects (OM)*: The defects will result in open faults if the spot defects break the connection patterns in one layer. Such a case where a spot defect breaks a net in the metal layer is shown in Fig. 1(b). If the defects cause missing via patterns, the open faults will also be induced because of missing vias.
3) *Inter-Layer extra oxide defects (IE)*: If the defects occur in the oxide at the location of vias, the vias may be blocked, thus leading to open faults. Fig. 2(a) shows an example where a via connecting metal 1 and metal 2 is broken by the spot defect.
4) *Inter-Layer missing oxide defects (IM)*: The defects-are also referred to as *oxide pinholes*. If the defects occur in the oxide between two overlapping conductors, the conductors are shorted. For instance, the pinhole in Fig. 2(b) causes a new via connecting metal 1 and metal 2, and therefore results in a bridge.

In the next section, assuming the above four types of spot defects to be the main random disturbance in the IC processes, we propose a formula to predict the probability of the failure of a net by taking into account the spot defect size distribution and the critical areas with respect to the various types of spot defects. Based on the formula, a routing strategy is suggested to minimize the probability of the failure of a net during the creation of the net.

III. The Failure Probability of a Net

Critical area, defined as the area in which the center of a defect must fall to cause a fault, can be extended to the critical area with respect to a particular *object*. The object can be any spot defect type. Suppose the spot defect size distribution for object η is $D_\eta(x)$, and the critical area with respect to object η is $A_\eta(x)$, where x is the spot defect size. If a uniform defect density P_η is assumed, then the probability of the failure of object η, denoted as F_η, can be expressed as:

$$F_\eta = P_\eta \int_{min}^{max} D_\eta(x) A_\eta(x)\, dx$$

where min and max are the minimum and the maximum defect sizes. There have been many efforts on modeling the defect size distribution. In this paper, the size distribution function taken from [10] is assumed. In principle, using other size distribution functions will not affect the following discussions. By replacing function $D_\eta(x)$ with X_0^2/x^3, where X_0 is the peak defect size of the distribution, we obtain

$$F_\eta = P_\eta X_0^2 \int_{min}^{max} A_\eta(x) \frac{1}{x^3}\, dx. \qquad (1)$$

As described in the previous section, the spot defects can be classified by four types. For each net, the critical area $A_\eta(x)$ with respect to the spot defects of type η can be estimated by using the virtual artwork concept proposed in [7]. Hence, the failure probability for each type of spot defect can be computed according to (1).

Given a net N, suppose the net length is l, and the net width and spacing are w and s, respectively. Assume b is the total length of the adjacent segments with the neighboring nets, and o is the number of overlapping sites, i.e., the number of unit area overlaps with the conductors in the upper or lower layer as shown in Fig. 3. In addition, we suppose the number of vias on net N is v.

1) *Type OE*: If the defect size x is smaller than s, then the defects will not cause any fault due to the zero critical area. If $s \le x < 2s+w$, the critical area $A_{\mathrm{OE}}(x)$ is equal to $(x-s)b$. However, when the defect size is equal or larger than $2s+w$, the critical area will be saturated to $(s+w)b$. Consequently, the probability of the failure of defect type OE is

$$F_{\mathrm{OE}} = P_{\mathrm{OE}} X_0^2 b \left(\int_s^{2s+w} \frac{x-s}{x^3}\, dx + \int_{2s+w}^{max} \frac{s+w}{x^3}\, dx \right).$$

By setting max to ∞, we obtain $F_{\mathrm{OE}} = \alpha b$, with

$$\alpha = \frac{P_{\mathrm{OE}}X_0^2}{2}\left(\frac{1}{s} - \frac{1}{(2s+w)}\right). \tag{2}$$

2) *Type OM*: When the defect size is smaller than w, it is not possible that the net will be broken by the defect. Therefore, the critical area is zero. For the defects with size $x, w \leq x < 2w + s$, the critical area $A_{\mathrm{OM}}(x)$ is equal to $(x - w)l$. As the defect size exceeds $2w + s$, the critical area will be saturated to $(w + s)l$, similar to the defect type OE. Consequently the probability of the failure caused by the defects of type OM can be described as

$$F_{\mathrm{OM}} = P_{\mathrm{OM}}X_0^2 l\left(\int_w^{2w+s} \frac{x-w}{x^3}\,dx + \int_{2w+s}^{\infty} \frac{s+w}{x^3}\,dx\right).$$

Similarly, we derive $F_{\mathrm{OM}} = \beta l$, where

$$\beta = \frac{P_{\mathrm{OM}}X_0^2}{2}\left(\frac{1}{w} - \frac{1}{(2w+s)}\right). \tag{3}$$

3) *Type IE*: Since the defects of type IE will only break conductors traversing the oxide, i.e., vias, the probability of the failure caused by this type of defects is proportional to the number of vias on the net. It needs to be mentioned that the defects in oxide no longer follow the normal size distribution function. Here, we assume a simple model to estimate the failure probability. (Since the critical area for vias given defect size x is a function of x^2, (1) will yield an infinite value for γ if the size distribution function of [10] is used.) Suppose the size of a via is $w \times w$. The probability of failure can be estimated by $F_{\mathrm{IE}} = \gamma v$, where

$$\gamma = P_{\mathrm{IE}}w^2. \tag{4}$$

4) *Type IM*: The defects will cause parasitic vias between two layers of the silicon structure. However, the parasitic vias are functionally harmful only if the vias occur in places where two conductors overlap. As a result, the conductors are shorted by the pinhole defects. The overlap area can be treated as the critical area for the pinhole defects, assuming: 1) that a pinhole occurring in the overlap area will result in a bridge fault, and 2) there is no size distribution for pinhole defects. Therefore, the probability of failure caused by defects of type IM can be estimated by $F_{\mathrm{IM}} = \delta o$, where

$$\delta = P_{\mathrm{IM}}w^2. \tag{5}$$

According to the above analysis, the parameters $P_{\mathrm{OE}}, P_{\mathrm{OM}}, P_{\mathrm{IE}}, P_{\mathrm{IM}}$, and X_0 are process-related, while w and s are determined by the design rules. Since these parameters are independent to routers, the total probability of the failure F of net N can be given by summing up the probabilities of the failures caused by the different types of defects, i.e.,

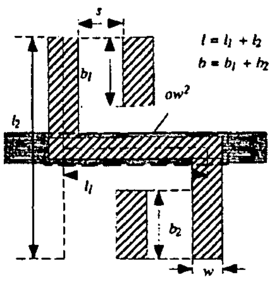


Fig. 3. Explanation of parameters w, s, b, l, o.

$$\begin{aligned} F &= F_{\mathrm{OE}} + F_{\mathrm{OM}} + F_{\mathrm{IE}} + F_{\mathrm{IM}} \\ &= \alpha b + \beta l + \gamma v + \delta o \end{aligned} \tag{6}$$

where α, β, γ, and δ are given by the previous equations. It is obvious that the reduction of b, l, v, and o is an effective way to decrease the probability of the failure of net N for a router.

IV. New Cost Function

Given a routing graph $G = (V, E)$ with edge weights $\lambda(e)$. The cost of a net is defined as the sum of the cost of the edges of the Steiner tree that connects the terminal vertices. Let $E_T \subseteq E$ denote the set of edges; then, the cost of a net is given by

$$C = \sum_{e \in E_T} \lambda(e). \tag{7}$$

The goal is to find a minimum cost connection for each net. We combine the conventional cost function of (7) with the failure cost function of (6) according to

$$C_{\mathrm{new}} = C + \rho F. \tag{8}$$

In conventional routing algorithms, the goal is to achieve minimum total net length, implying minimum area. Thus, the conventional cost function is modeled as a minimum length cost function. In addition to the net length and the number of vias which are considered in conventional cost functions, the failure cost function introduces two new aspects, namely bridges and overlaps. In essence, minimizing both net length and bridges/overlap is contradictory. Therefore, for dense circuits, net length minimization should be favored over minimizing bridges/overlap because routing space is limited, as opposed to sparse circuits, where minimization of bridges/overlap may be favored over net length minimization. Thus, ρ is directly proportional to the sparsity of a circuit. We define the sparsity of a circuit as

$$s = 1 - \frac{A_n}{A_r} \tag{9}$$

where A_n denotes the amount of space necessary to lay down all nets as estimated by the global router and A_r denotes the amount of free routing space after placement. Notice that

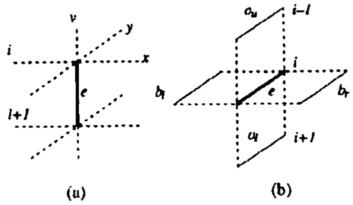


Fig. 4. Grid model.

maximally sparse circuits have $s = 1$ and maximally dense circuits have $s = 0$. Obviously, $s < 0$ indicates circuits that are not routable.

Since ρ is a weight factor, it depends on the actual values occurring in the conventional cost function C. As we will show in the next section, we can derive a weight factor σ to take into account this dependency. Thus, we may write ρ as

$$\rho = s\sigma. \tag{10}$$

V. Incorporating Routing Style

We assume that the routing space is modelled as a 3-D grid graph $G = (V, E)$. An edge $e \in E$ of the grid graph may have one of three directions, called x-, y-, and *v-directions*, as indicated in Fig. 4(a). Vias are represented by edges in the v-direction. Wires are allowed to run over edges and bend at grid points. An edge $e \in E$ of the grid graph is said to be *active* if it is part of a wiring pattern. Edges that are not part of a wiring pattern are called *inactive*. The status of an edge may be changed from inactive to active by the router. Possibly, initial wiring patterns exist in the grid graph.

As mentioned in the introduction, we distinguish three aspects that may affect the edge cost function $\lambda(e)$. To cover these aspects, we assume that for each layer i, three costs are specified, namely c_i^x, c_i^y, and c_i^v. Here, c_i^x denotes the cost of edges in the x-direction, c_i^y denotes the cost of edges in the y-direction, and c_i^v denotes the cost of vias connecting layer i and $i + 1$.

Let $l_i = l_i^x + l_i^y$ denote the total number of edges in layer i for a net, where l_i^x and l_i^y denote the number of edges in the x- and y-directions, respectively. Furthermore, let v_i denote the number of vias connecting layers i and $i + 1$. Then, we may write (7) as

$$C = \sum_i c_i^x l_i^x + c_i^y l_i^y + c_i^v v_i. \tag{11}$$

Since the failure cost function is specific to some material, we assume that for each layer i a failure cost function according to (6) is specified, i.e., $F = \alpha_i b_i + \beta_i l_i + \gamma_i v_i + \delta_i o_i$. Then, combining the conventional cost function of (11) with the failure cost function according to (8) yields

$$C = \sum_i (c_i^x + \rho_i\beta_i)l_i^x + (c_i^y + \rho_i\beta_i)l_i^y$$
$$+ (c_i^v + \rho_i\gamma_i)v_i + \rho_i\alpha_i b_i + \rho_i\delta_i o_i. \tag{12}$$

Since the cost of vias are not influenced by any existing wiring pattern, we may discard vias from the following discussion, and set the cost of a via in layer i to $c_i^v + \rho_i\gamma_i$.

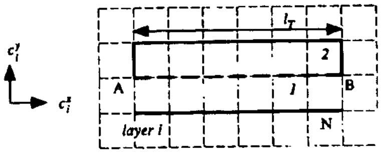


Fig. 5. Determination of cost using the new cost function. When minimizing bridges, variant 2 is preferred to variant 1.

Assume that ρ is specified for each layer according to $\rho_i = s\sigma_i$. Furthermore, assume that the circuit is maximally sparse, i.e., $s = 1$, implying that $\rho_i = \sigma_i$. Since the circuit is maximally sparse, we want to minimize bridges/overlap.

In Fig. 5, a net N exists in the routing space. Connecting point A and point B, we want the net to follow variant 2 instead of 1, because the critical area for bridges is minimal for variant 2. Using (12) and assuming the length of the net in the x-direction is given by l_T, the cost of both variants are given by

$$C_1 = (c_i^x + \sigma_i\beta_i)l_T + \sigma_i\alpha_i l_T$$
$$C_2 = (c_i^x + \sigma_i\beta_i)l_T + 2(c_i^y + \sigma_i\beta_i)$$

using $b = l_T$ for variant 1 and $b = 0$ for variant 2. Since we prefer variant 2 to variant 1, we demand that $C_1 > C_2$ and derive a lower bound for σ_i, i.e.,

$$\sigma_i > \frac{2c_i^y}{\alpha_i l_T - 2\beta_i}. \tag{13}$$

Similarly, for vertical wires we derive

$$\sigma_i > \frac{2c_i^x}{\alpha_i l_T - 2\beta_i}. \tag{14}$$

For overlap we may derive the same functions, only substituting δ_i for α_i, i.e.,

$$\sigma_i > \frac{2c_i^x}{\delta_i l_T - 2\beta_i} \quad \text{and} \quad \sigma_i > \frac{2c_i^y}{\delta_i l_T - 2\beta_i}. \tag{15}$$

Combining (13)–(15), and setting σ_i to the maximum lower bound yields

$$\sigma_i = \frac{2\max(c_i^x, c_i^y)}{l_T \min(\alpha_i, \delta_i) - 2\beta_i}. \tag{16}$$

As can be seen from (16), σ_i depends on both the cost information per layer and the failure parameters specific to each layer. Parameter l_T may be seen as a threshold net length. If the length by which two nets are in parallel (or overlap) exceeds this threshold, we demand that one of the nets will take a detour as shown in Fig. 5.

VI. Computing New Edge Cost

A procedure is given in this section to determine the final cost of an edge (see Algorithm 1); to be able to do this, the notion of *surrounding edges* is introduced. For each edge $e \in E$ in the x- or y-directions, four surrounding edges are identified, denoted as b_l, b_r, o_u, and o_l as indicated in Fig. 4(b). The edges b_l and b_r, lying in the same layer as edge e, form the possible bridging edges, while o_u and o_l, lying in the upper and lower layers, respectively, form possible overlap edges.

```
procedure determine_edge_cost $(i, dir)$
begin
    if $dir = v$ then $\lambda := c_i^v + \rho_i \gamma_i$;
    else
        $\lambda := c_i^{dir} + \rho_i \beta_i$;
        if $b_l$ is active then $\lambda := \lambda + \rho_i \alpha_i$;
        if $b_r$ is active then $\lambda := \lambda + \rho_i \alpha_i$;
        if $o_u$ is active then $\lambda := \lambda + \rho_i \delta_i$;
        if $o_l$ is active then $\lambda := \lambda + \rho_i \delta_{i-1}$;
    fi;
    return $\lambda$;
end
```

A. Algorithm 1: Determination of the New Edge Cost

In the above procedure, i is the index of the layer in which edge e lies, and dir denotes the direction of the edge, being either x, y, or v. No special actions are taken for vias; if the edge represents a via from layer i to layer $i + 1$, the cost λ is set to $c_i^v + \rho_i \gamma_i$. The final cost is influenced by their surrounding active edges for all other edges. The edge e is assigned the original cost $c_i^{dir} + \rho_i \beta_i$, plus a cost for each active surrounding edge. The latter depends on the relative position of the surrounding edge with respect to edge e. It is easy to see that this procedure assigns the original edge cost if $s = 0$, implying $\rho_i = 0$, for all i.

The final edge cost depends entirely on the active edges by which it is surrounded, and therefore may change during routing. To avoid changes in cost due to interaction with already routed segments of the same net, it is assumed that an edge is activated only after all terminals of a net are connected. Notice that the above procedure takes constant time to determine the cost of an edge. Therefore, the run time complexity of the original maze router is not influenced by this new cost function.

VII. Experiments

The routing approach in which the layout failure mechanism is taken into account has been implemented in the GAS sea of gates layout system [11], using the multiterminal maze router of [3]. To test the real effect of our new routing strategy, 20 different circuits have been laid out. Except for mult8 and primes9, all circuits are taken from the MCNC '91 logic synthesis benchmark set. The scales of the layouts range from 150 to 5000 transistors, while their numbers of nets range from 100 to 3500. After placement is finished for each circuit, the sparsity of a layout can be obtained according to (9). Basic information about the benchmark layouts as well as their sparsities is shown in Table I.

All benchmark circuits are routed using the detailed router of the GAS system. To compare the results achieved by the newly proposed cost function, each circuit is laid out twice, once using the conventional routing cost function and once using the new routing cost function. Routing is performed using three layers: a polysilicon layer *ps*, and two metal layers *in* and *ins*. The original edge costs are set according to $c_{ps}^x = 20$, $c_{ps}^y = 3$, $c_{in}^x = 3$,$c_{in}^y = 10$, $c_{ins}^x = 8$, and $c_{ins}^y = 2$, imposing a vertical–horizontal–vertical (VHV) routing style. Without loss of generality, the parameters α, β, γ, δ are set to 1, and l_T is set to 7. Consequently, for each of the three layers σ can be obtained, i.e., $\sigma_{ps} = 8$, $\sigma_{in} = 4$, and $\sigma_{ins} = 3$.

TABLE I
Circuit Statistics and Runtimes

circuit	# trans	# nets	s (%)	runtime conv. (sec)	runtime new (sec)	runtime increase (%)
alu2	372	206	78	4.0	5.9	47.5
apex3	4894	3283	48	766.9	875.8	14.2
apla	540	290	77	11.4	13.6	19.3
bw	504	267	76	14.3	16.8	17.5
clip	446	242	78	11.7	13.9	18.8
dk17	328	182	80	7.8	9.6	23.1
duke2	1206	641	67	77.6	90.8	17.0
e64	700	429	86	8.0	10.7	33.7
5xp1	332	181	80	4.1	4.8	17.1
9sym	658	350	71	2.0	2.5	25.0
in6	817	455	73	20.2	24.5	21.3
misex2	564	316	78	15.3	17.8	16.3
mult8	1566	837	76	37.5	45.1	20.3
o64	572	428	86	8.2	9.7	18.3
primes9	2112	1167	62	202.2	234.2	15.8
radd	146	87	86	1.1	1.1	0.0
rd84	547	290	75	6.0	7.8	30.0
sao	526	283	72	9.0	10.5	16.7
six	358	193	75	5.6	7.3	30.4
vg2	245	177	84	6.0	7.6	26.7

The run times are presented for both runs of the router, respectively, using the conventional routing cost function and the new routing cost function (Table I). Experiments were done on a HP735. On average run time increases by 21.4%. It is clear that this increase in run time comes from the determination of the edge cost during routing. However, this determination still takes constant time, and thus the complexity of the routing algorithm is not changed.

The EDAM system [13] is used to obtain data concerning the failure probability of both layouts. According to (1), the failure probability of a layout largely relies on the values of parameters P_η and X_0, which are process-environment dependent. Thus, probability computation will not make sense without accurate values for these parameters. In this paper, we make the simplification of computing the layout sensitivity instead of the failure probability, because it is believed that a low layout sensitivity implies a small failure probability. A defect size of 4 μm is chosen to evaluate the critical areas of the benchmark layouts, whose feature size is scaled down to 2 μm. Therefore, the defect size is large enough to reflect meaningful layout sensitivities.

The critical areas with respect to the four types of faults are computed for each circuit. The changes in the critical areas as well as the layout sensitivities (ΔLS) are presented in Table II. From the data, it may be concluded that for all benchmark layouts the critical areas for one layer bridge faults (type OE) decrease 22.6% on average, while the critical areas with respect to one layer open faults only increase 2.5% on average. The critical areas for inter-layer faults, i.e., type IE faults and type IM faults, change very slightly. For the IE faults, this is because via-minimization is already considered in the original cost function. Therefore, the number of vias will slightly increase since the weight of a via is relatively small in

TABLE II
ANALYSIS RESULTS

circuit	change of crit. area (%)				ΔLS
	OE	OM	IE	IM	(%)
alu2	−23.2	3.9	6.9	2.4	−5.0
apex3	−16.4	0.3	3.0	−7.8	−6.4
apla	−23.3	0.6	5.0	2.5	−7.5
bw	−27.2	3.6	9.7	2.4	−7.8
clip	−24.4	1.0	3.9	3.4	−7.7
dk17	−20.0	3.8	4.2	1.8	−4.2
duke2	−22.1	2.1	4.3	1.3	−6.8
e64	−13.8	0.7	0.0	0.3	−4.0
5xp1	−30.2	2.5	6.1	1.5	−8.8
9sym	−27.6	3.6	5.6	4.3	−7.5
in6	−38.2	4.5	1.2	2.6	−7.9
misex2	· 26.7	1.9	4.2	1.7	−7.9
mult8	−21.4	2.5	3.3	−5.5	−7.5
o64	−14.2	0.6	6.3	−0.1	−4.9
primes9	−16.1	3.8	10.5	−8.6	−4.4
radd	−15.5	3.8	12.6	0.5	−3.5
rd84	−25.2	4.0	4.1	2.7	−6.7
sao	−24.3	2.6	3.1	3.5	−7.3
six	−23.4	3.6	10.4	−0.8	−6.2
vg2	−17.8	0.5	3.2	1.7	−5.2

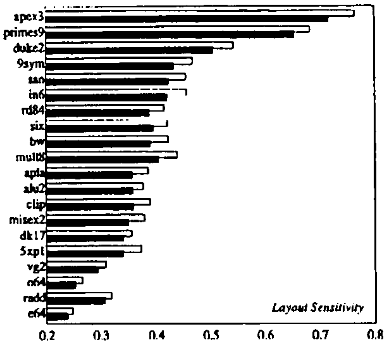


Fig. 6. Change in layout sensitivity per circuit.

the new cost function. The changes of the fault IM seem to be random. The reasons are as follows: 1) In the original routing module, the VHV routing style is chosen. Therefore, large area overlap between two metal layers is already prevented by the design style. Obviously, the critical areas caused by the overlaps will not be decreased significantly by putting an extra penalty on them. 2) Since in some cases an increase in overlap between two layers will result in a final decrement of the total cost, it is also possible that the critical areas with respect to these faults will increase.

The total effect of the new routing strategy is shown in the last column in Table II. According to the data, we find that the layout sensitivities can be decreased 6.4% on average, if the failure probability is taken into account in the routing procedure.

Fig. 6 shows the sensitivities of the two different layouts per design: the white bars represent the sensitivities of the layouts made by the original router and the black bars indicate the sensitivities of the layouts made by the new routing module.

To give an indication of the effect the new cost function on layout, a snapshot is taken from the layout of benchmark circuit apla. Fig. 7 shows the layout obtained using the conventional cost function and Fig. 8 shows the layout after the new cost function is used. Clearly, the wiring on all three routing layers is spread more uniform over the available area, and the amount of overlap between wires on different layers is less in Fig. 8. Both the number of vias and the net length increase slightly for Fig. 8.

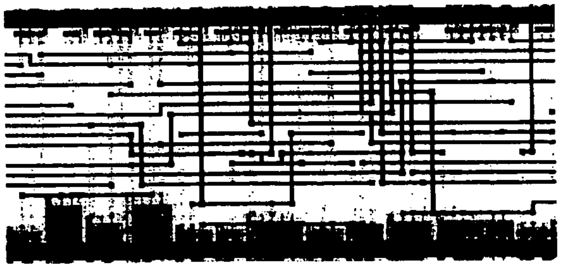


Fig. 7. Snapshot of layout of circuit 'apla' using conventional routing cost function.

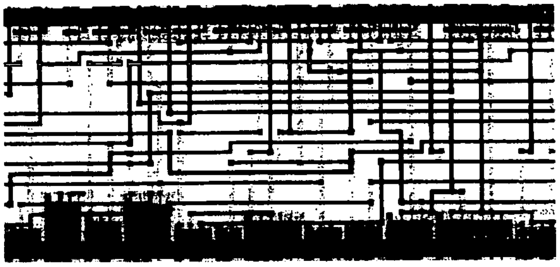


Fig. 8. Snapshot of layout of circuit 'apla' using the new routing cost function.

VIII. CONCLUSIONS

A novel routing strategy producing layouts that are less susceptible to spot defects has been presented. Based on an analysis of spot defects, the four types of the main random disturbance in IC processes are modelled. A formula indicating the failure probabilities of these faults is derived. Combining the failure cost function with the conventional cost function, a new cost function for the general routing problem is devised. By using this new cost function, a good tradeoff between the minimization of the total net length and the maximization of the manufacturability of a layout can be obtained. The experimental data show that the layout sensitivities can be significantly decreased by the proposed routing approach even for very dense circuits, while the layout areas are kept the same.

REFERENCES

[1] K. Balachandran *et al.*, "A yield enhancing router," in *Proc. Int. Symp. Circuits Syst.*, 1991, pp. 1936–1939.

[2] V. K. R. Chilivuri and I. Koren, "New routing and compaction strategies for yield enhancement," in *IEEE Int. Workshop on Defect and Fault Tolerance*, 1992, pp. 325–334.

[3] E. P. Huijbregts and J. A. G. Jess, "General gate array routing using a *k*-terminal net routing with failure prediction," *IEEE Trans. VLSI Syst.*, vol. 1, no. 4, Dec. 1993.

[4] F. K. Hwang and D. S. Richards, "Steiner tree problems," *Networks*, vol. 22, no. 1, pp. 55–89, Jan. 1992.

[5] S. Y. Kuo, "YOR: A yield optimizing routing algorithm by minimizing critical areas and vias," *IEEE Trans. Computer-Aided Design*, vol. 12, no. 9, pp. 1303–1311, Sept. 1993.

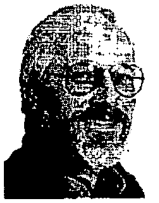
[6] T. Lengauer, *Combinatorial Algorithms for Integrated Circuit Layout.* New York: Wiley, 1990.
[7] W. Maly, "Modeling of lithography related yield losses for CAD of VLSI circuits," *IEEE Trans. Computer-Aided Design*, vol. CAD-4, no. 3, pp. 166–177, July 1985.
[8] A. Pitaksanonkul *et al.*, "DTR: A defect-tolerant routing algorithm," in *Proc. Des. Automat. Conf.*, 1989, pp. 795–798.
[9] J. P. Shen, W. Maly, and F. J. Ferguson, "Inductive fault analysis of MOS integrated circuits," *IEEE Design, Test*, vol. 2, pp. 13–26, Dec. 1985.
[10] C. H. Stapper, "Modeling of defects in integrated circuit photolithographic patterns," *IBM J. Res. Develop.*, vol. 28, no. 4, pp. 461–475, July 1984.
[11] A. G. J. Slenter, "A generalized approach to gate array layout design automation," Ph.D. dissertation, Eindhoven University of Technology, The Netherlands, 1990.
[12] A. J. Strojwas, "Design for manufacturability and yield," in *Proc. Des. Automat. Conf.*, 1989, pp. 454–459.
[13] H. Xue, C. N. Di, and J. A. G. Jess, "A net-oriented method for realistic fault analysis," in *Proc. IEEE Int. Conf. Computer-Aided Design*, 1993, pp. 78–84.
[14] H. Xue, E. P. Huijbregts, and J. A. G. Jess, "Routing for manufacturability," in *Proc. Des. Automat. Conf.*, 1994, pp. 402–406.

Ed P. Huijbregts was born on December 4, 1965, in Breda, The Netherlands. He received the M.S. degree in electronic engineering from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1989.

From 1989 to 1994 he worked towards a Ph.D. degree in the Design Automation Section at the Eindhoven University of Technology, Eindhoven, The Netherlands. His work there included the development of placement and routing tools for flexible macros. He joined Philips Research Laboratories, Eindhoven, The Netherlands, in February 1994, where he holds the position of research scientist. He is currently working on power estimation tools for VLSI; his interests are in the areas of VLSI design and layout, and applied graph theory.

Hua Xue received the B.S., M.S., and Ph.D. degrees in electronic engineering from the Fudan University, China in 1985, 1988, and 1991, respectively.

He worked on the PANDA project in the Beijing IC Design Center from 1988 to 1989. From 1991 to 1992, he was a lecturer at the Electronics Engineering Department of Fudan University, where he worked on the development of a performance-driven gate array layout system. From 1992 to 1994, he was a research fellow at the Design Automation Section at Eindhoven University of Technology, The Netherlands. He joined Xilinx Inc., San Jose, CA, in 1994, where is a senior software engineer. His work there included realistic fault analysis and IC test strategy. His current research interests include logic synthesis and performance-driven technology mapping.



Jochen A. G. Jess was born on April 13, 1935, in Dortmund, Germany. He received the M.S. degree from the Rheinisch-Westfalishe Technische Hochschule Aachen, Germany, 1960 and the Ph.D. degree from the Aachen University of Technology, Germany in 1963.

From 1963 to 1968 he was a research staff member at the Institut fur Nachrichtensysteme, Kalsruhe University of Technology. From 1968 to 1969 he was a Visiting Professor at the Department of Electrical Engineering, University of Maryland, College Park, Maryland. From 1969 to 1971 he was a senior staff member at the Kalsruhe University of Technology. Since 1971 he has been Professor and Head of the Design Automation Section at the department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands. His current interests are in the design and design automation of integrated circuits, in particular layout design, logic design, design of architectures and formal verification. He is co-author of more than 75 papers. He is chairman of the European Design and (Design) Automation Association (EDAA) and served as program chairperson of ICCAD93.

IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark Office

Help FAQ Terms IEEE Peer Review

Quick Links

» **Abstract Plus**

Welcome to IEEE Xplore®

- ○ Home
- ○ What Can I Access?
- ○ Log-out

Tables of Contents

- ○ Journals & Magazines
- ○ Conference Proceedings
- ○ Standards

Search

- ○ By Author
- ○ Basic
- ○ Advanced

Member Services

- ○ Join IEEE
- ○ Establish IEEE Web Account
- ○ Access the IEEE Member Digital Library

Print Format

SEARCH RESULTS [PDF Full-Text (284 KB)] PREVIOUS NEXT DOWNLOAD CITATION

Partitioning logic to optimize routability on graph structures

Vijayan, G.
IBM Thomas J. Watson Res. Center, Yorktown Heights, NY;

This paper appears in: **Circuits and Systems, 1990., IEEE International Symposium on**
Meeting Date: 05/01/1990 -05/03/1990
Publication Date: 1-3 May 1990
Location: New Orleans, LA , USA
On page(s): 2638-2641 vol.4
References Cited: 9
INSPEC Accession Number: 3887905

Abstract:
The problem of partitioning the nodes of a logic network (i.e. a hypergraph) on to the vertices of a partition graph G, in which the cost function to be minimized is the cost of global routing (i.e. the cost of routing the nets of the logic on the edges of the graph G), is studied. Each vertex of the partition graph has a given upper bound on the number of nodes of the logic that can be assigned to the vertex. the nets of the logic network and the edges of the partition graph may have weights associated with them, which appear as multiplicative factors in the routing cost function. This partitioning program is called the min-cost partitioning on a graph (MCPG) problem. The MCPG model is very general and can be applied in many partitioning situations arising in VLSI physical design. Two such applications are described

Index Terms:
VLSI graph theory logic design MCPG model VLSI physical design cost function global routing graph structures hypergraph logic network min-cost partitioning on a graph multiplicative factors nodes partition graph partitioning routability routing cost function upper bound weights

Documents that cite this document
Select link to view other documents in the database that cite this one.

SEARCH RESULTS [PDF Full-Text (284 KB)] PREVIOUS NEXT DOWNLOAD CITATION

Home | Log-out | Journals | Conference Proceedings | Standards | Search by Author | Basic Search | Advanced Search
Join IEEE | Web Account | New this week | OPAC Linking Information | Your Feedback | Technical Support | Email Alerting
No Robots Please | Release Notes | IEEE Online Publications | Help | FAQ| Terms | Back to Top

Partitioning Logic to Optimize Routability on Graph Structures

Gopalakrishnan Vijayan
IBM Research Division
T. J. Watson Research Center
Yorktown Heights, NY 10598

1. Introduction

We study the problem of partitioning the *nodes* of a logic network (i.e., a hypergraph) on to the *vertices* of a *partition graph G*, in which the cost function to be minimized is the cost of global routing, i.e., the cost of routing the nets of the logic on the edges of the graph *G*. Each vertex of the partition graph has a given upper bound on the number of nodes of the logic that can be assigned to the vertex. The *nets* of the logic network and the *edges* of the partition graph may have weights associated with them, which appear as multiplicative factors in the routing cost function. We refer to this partitioning problem as the **Min-Cost Partitioning on a Graph (MCPG)** problem.

The MCPG model generalizes many different formulations of logic partitioning [1,3,4,6,8,9]. For example the classical min-cut problem of [1,3,4] reduces to the case where the partition graph *G* is just a single edge (see Figure 1). The MCPG model is very general and can be applied in many partitioning situations arising in VLSI physical design. We describe two such applications:

Residual Logic Partitioning: In many practical chip designs there are usually a set of macros such as memory blocks or data-path stacks, which have to be pre-placed due to timing, I/O proximity, power/ground bussing or other considerations. The residual area on the chip may be highly non-rectangular and contain many bottlenecks for wiring. This residual area has to be populated with the residual logic (usually control logic). A suitable approach is to dissect the residual area into rectangular regions, partition the residual logic on to these regions, and then perform placement within each region. The MCPG model can be used in the partitioning step in order to optimize the number of nets that cross the cuts separating the rectangular regions. The partition graph to be used will be the planar dual of the rectangular dissection.

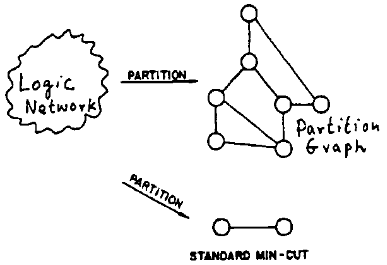


Figure 1. The MCPG and standard min-cut models

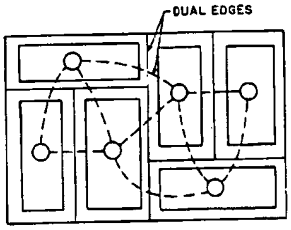


Figure 2. Partition Graph (the dual) of a Floorplan

Partitioning after Early Floorplanning: In timing-critical VLSI designs, it is desirable and many times imperative, to influence the logic design process using physical design constraints [2]. One such method is to do *early floorplanning* using the functional blocks and their connectivity, before the synthesis of the logic within the functional blocks. The floorplan can then be used to drive the logic synthesis process, so as to minimize delays on critical paths that traverse across many blocks of the floorplan. The synthesis of the functional blocks actually gives a initial and rough partition of the logic on to the blocks of the floorplan, which satisfies the timing constraints. Global wirability can then be improved by perturbing the functional partition. The MCPG model can be used to achieve this by re-mapping the logic on to a partition graph which is the the planar dual of the floorplan (Figure 2).

We now briefly describe a strategy for the modeling and solution of the MCPG problem. Given a partition, the feasibility and cost evaluation of global routing on the partition graph can be modeled as a multicommodity flow problem [7], which in turn has an equivalent Linear

CH2868-8/90/0000-2638$1.00 © 1990 IEEE

Programming (LP) formulation. The LP cost function is the weighted sum of flows on the edges of the partition graph, which is the global routing cost. In order to find a partition that minimizes this cost, the LP model has to be invoked and solved repeatedly within the framework of a partitioning algorithm which maps logic on to the vertices of a graph. However, the LP model is too large, and its solution too slow, to be used within such a partitioning program. Fortunately, there is a *necessary condition* for feasible multicommodity flow called the *cut condition* [5], which can be used to derive a suitable *net cost model* for the nets. The net cost model can be used to drive any partitioning heuristic, such as an iterative improvement heuristic for multiway partitioning [9] or a randomized heuristic. The partition obtained using the heuristic is verified for feasibility of global routing on the partition graph using the LP model. If feasibility is not attained, the weights on the edges of the partition graph are modified appropriately to reflect the new criticalities of the edges. This results in a change in the parameters in the cost function used by the partitioning heuristic. The partitioning heuristic is reapplied, and this process is repeated until a feasible global routing is attained. The LP procedure is thus pushed to the outer loop of the overall partitioning strategy, with a cost model given by the cut condition being used within the inner loop.

2. A Net Cost Model

Given a graph $G=(V,E)$ and a susbet $S \subseteq V$, we denote by G/S the subgraph of G which is *induced* by the subset S. We denote the complement $V-S$ of the set S with respect to V as $\bar{S}$ A *cut* of the graph G, denoted as $(S,\bar{S})$ is defined by any subset $S \subseteq V$. Define $conn\_cuts(G)$ to be the collection of all cuts $(S,\bar{S})$ such that G/S and $G/\bar{S}$ are both *connected subgraphs*. In all references to a cut $(S,\bar{S})$ in the following discussion, we assume that $(S,\bar{S}) \in conn\_cuts(G)$.

Given a set $S \in conn\_cuts(G)$, we define $cross(S,\bar{S}) = \{(i,j) \in E \mid i \in S, j \in \bar{S}\}$, which is the set of edges of G that go between S and $\bar{S}$. The total capacity of the cut $(S,\bar{S})$ is given by $c(S,\bar{S}) = \sum_{e \in cross(S,\bar{S})} c_e$, where c_e denotes the capacity of edge e.

Given a graph G for the multicommodity flow problem, and a cut $(S,\bar{S})$, define $f(S,\bar{S})$ to be the total amount of flow that must cross this cut.

The **cut condition** [5] is defined to be $f(S,\bar{S}) \leq c(S,\bar{S})$; $\forall S \in conn\_cuts(G)$.

From its definition, it is clear that the *cut condition* is a *necessary condition* for the existence of a feasible multicommodity flow.

We now describe a suitable cost model for nets that is inspired by the cut condition. Let $L=(U,H)$ be a logic network, where U is its set of nodes and H its collection of nets. Consider a partition M of the node set U on to the vertex set V of the partition graph G. Let $M(u)$ denote the the vertex of V containing the node u under the partition M . For a net $h \in H$, let $M(h) = \{ M(u) \mid u \in h \}$, i.e., each vertex in $M(h)$ contains at least one node of the net h. The net h is said to *contribute* to a cut $(S,\bar{S})$, if $M(h) \cap S \neq \phi$ and $M(h) \cap \bar{S} \neq \phi$. In other words, the cut $(S,\bar{S})$ cuts the span of the net h, and the net h must flow across the cut $(S,\bar{S})$. Define the set $span\_cuts(M(h))$ to be the collection of cuts $(S,\bar{S}) \in conn\_cuts(G)$ such that the net h contributes to the cut $(S,\bar{S})$, i.e.,

$span\_cuts(M(h)) =$
$\{(S,\bar{S}) \in conn\_cuts(G) \mid M(h) \cap S \neq \phi, M(h) \cap \bar{S} \neq \phi\}$

A suitable *cost* that the cut condition suggests for the net h under a given partition M is the cardinality of the set $span\_cuts(M(h))$. Any reduction in the number of these contributed cuts for a net is a positive step towards the satisfaction of the cut condition. However, this cost model does not take into account the weights of the edges belonging to a cut, nor the weight of the net itself.

Let w_e denote the weight assigned to an edge e of the partition graph. Let us examine how to assign a suitable *weight* to a cut $(S,\bar{S})$ using the weights of the edges in $cross(S,\bar{S})$. To take a simple example, let there be just two edges with equal weight 10 in $cross(S,\bar{S})$. Intuitively, the weight of the cut should be less than the weight of both the edges, because two edges are better than any one edge as far as flow across cut is considered. The *harmonic mean* $1/(1/10+1/10)=5$ has this desired property, and is a good measure for the weight of a cut.

Generalizing, we define the weight of a cut $(S, \bar{S})$ as $w(S, \bar{S}) = 1/(\sum_{e \in cross(S, \bar{S})} 1/w_e)$.

Let w_h denote the weight of a net h. The cost of the net h, under a partition M, denoted as *net_cost*(h) can be defined as

$$net\_cost(h) = w_h \times \sum_{S \in span\_cuts(M(h))} w(S, \bar{S}) .$$

Note that when all the nodes belonging to a net reside in the same vertex of G, i.e., when $|M(h)| = 1$, we have $span\_cuts(M(h)) = \phi$, and therefore $net\_cost(h) = 0$. Such a local net does not contribute to the cost of global routing.

The cost of a partition M is defined to be $cost(M) = \sum_{h \in H} net\_cost(h)$, i.e., the sum of the costs of the nets of the logic network L.

An Example:

Consider the partition graph shown in Figure 3. Let the weights of all 5 edges be equal to 1. Let h_1 be a net whose spanning set is $M(h_1) = \{1,3\}$. Let h_2 be a net whose spanning set is $M(h_2) = \{2,4\}$.

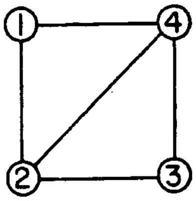


Figure 3. A partition graph

$$\begin{aligned}
span\_cuts(M(h_1)) &= \{ (\{1,2,4\},\{3\}), (\{1\},\{2,3,4\}), \\
&\quad (\{1,2\},\{3,4\}), (\{1,4\},\{2,3\}) \} \\
cross(\{1,2,4\},\{3\}) &= \{(4,3),(2,3)\} ; \\
w(\{1,2,4\},\{3\}) &= 1/(1+1) = 1/2 \\
cross(\{1\},\{2,3,4\}) &= \{(1,2),(1,4)\} ; \\
w(\{1\},\{2,3,4\}) &= 1/(1+1) = 1/2 \\
cross(\{1,2\},\{3,4\}) &= \{(1,4),(2,4),(2,3)\} ; \\
w(\{1,2\},\{3,4\}) &= 1/(1+1+1) = 1/3 \\
cross(\{1,4\},\{2,3\}) &= \{(1,2),(4,2),(4,3)\} ; \\
w(\{1,4\},\{2,3\}) &= 1/(1+1+1) = 1/3
\end{aligned}$$

$$net\_cost(h_1) = 1/2 + 1/2 + 1/3 + 1/3 = 5/3$$

Similarly, we can derive $net\_cost(h_2) = 4/3$

Intuitively, one can observe that there are more routes for the net h_2 than for the net h_1, and thus h_2 has a lower cost.

3. MCPG Partitioning Framework

Given a logic network L and a partition graph G, our framework for the MCPG solution is as follows:

```
1. Select an initial partition $M$ of the nodes of $L$ on to
   the vertices of $G$.
2. Select an appropriate initial weight $w_e$ for each edge
   $e$ of $G$.
3. Repeat the following steps
   a. Execute a partitioning heuristic, which uses
      $cost(M)$ as its cost function, and improve the
      partition $M$.
   b. Formulate the global routing LP model for the
      partition $M$ on the graph $G$, and check if the LP
      is feasible.
   c. If the LP is infeasible, then
         Appropriately adjust the weight $w_e$ of each edge
         $e$ of $G$.
   until the LP is feasible
```

Thus the LP model is thus pushed to the outer loop of the overall partitioning strategy, with a faster heuristic using the net cost model being used within the inner loop.

The Partitioning Heuristic: The *partitioning heuristic* of step 3(a) may be any suitable heuristic for the MCPG problem. One such multiway partitioning heuristic is described in [9].

Adjustment of the weights w_e: Edges with smaller capacity are bottlenecks for global routing. Flows on these edges should have a higher penalty associated with them. Therefore the initial weight w_e of an edge e may be set inversely proportional to its capacity c_e, as $w_e = \frac{C}{c_e}$, where C is a suitable constant. The readjustment of the weights in step 3(c) can be done as follows: Temporarily set the capacity of each edge to ∞, and solve the LP again, using the LP cost function, which is the weighted sum of flows on the edges of the partition graph. Note that this modified LP is always feasible. Let f_e be the flow of nets on the edge e in the LP solution. The difference $f_e - c_e$ is a measure of the flow criticality of the edge e. A positive difference for an edge e implies that there is an overflow on the edge, and that its weight must be set higher. A negative difference implies that the edge is not critical with respect to global routing, and its weight may be lowered. The new weight w_e can be set as $w_e\ (new) = w_e\ (old) + C' \times (f_e - c_e)$, where C' is another suitable positive constant.

Number of repetitions: In practice, we repeat step 3, five times, irrespective of whether the LP becomes feasible or not. Even if the LP becomes feasible after the first execution of the partitioning heuristic, it may be possible to improve the partition, by readjusting the weights and repeating this process four more times. If the LP does not become feasible after five repetitions, it is up to the designer to decide whether this implies an inherent infeasibility or whether the process should be repeated with a modified partition graph. This crucial decision depends largely on the application. For example, in the early floorplanning application, the designer may try a different floorplan of the same set of functional blocks, which also satisfies the timing and other considerations used in the early floorplanning process. Alternately, the designer may decide to try a different functional partitioning.

4. Conclusions

The MCPG framework employing an iterative improvement partitioning heuristic was implemented and used successfully on several real designs in a variety of different applications. To give one example, the partitioning system was used in the design of a dense CMOS chip which had 23 big predesigned macros, with the residual logic having more than 11000 logic nodes and about 12000 nets. The application here was a mixture of residual logic partitioning and partitioning after early floorplanning. With the use of the MCPG framework, the average net length was reduced by about 20%, and the number of nets that could not be routed by an automatic routing program was reduced by about 60%. Similar results were observed for other chip designs which required the use of the general partition model. One important benefit of the MCPG model is that it gives a practical approach to addressing the global routing problem very early in the design process, thus speeding up the design cycle time.

References:

1. C. M. Fiduccia and R. M. Mattheyses, "A Linear-Time Heuristic for Improving Network Partitions," *Proceedings of 19th ACM/IEEE Design Automation Conference*, Las Vegas, Nevada, pp. 175-181, June 1982.
2. P. S. Hauge, R. Nair, and E. J. Yoffa, "Circuit Placement for Predictable Performance" *Digest of Technical Papers, ICCAD-87,*, pp. 88-91, November 1987.
3. B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," *Bell System Technical Journal*, Vol. 49, February 1970.
4. B. Krishnamurthy, "An Improved Min-Cut Algorithm for Partitioning VLSI Networks," *IEEE Transactions on Computers* Vol. C-33, No. 5, pp. 438-446, May 1984.
5. H. Okamura, "Multicommodity Flows in Graphs," *Discrete Applied Mathematics*, Vol. 6, pp. 55-62, 1983.
6. L. A. Sanchis, "Multiple-Way Network Partitioning," *IEEE Transactions on Computers*, Vol. 38, No. 1, pp. 62-81, January 1989.
7. E. Shragowitz and S. Keel, "A global router based on a multicommodity flow model," *INTEGRATION*, Vol. 5, pp. 3-16, 1987.
8. P. A. Suaris and G. Kedem, "Quadrisection: A New Approach to Standard Cell Layout," *Digest of Technical Papers, ICCAD-87,*, pp. 474-477, November 1987.
9. G. Vijayan, "Min-Cost Partitioning on a Tree Structure and Applications," *Proceedings of the 26th ACM/IEEE Design Automation Conference*, pp. 771-774, June 1989.